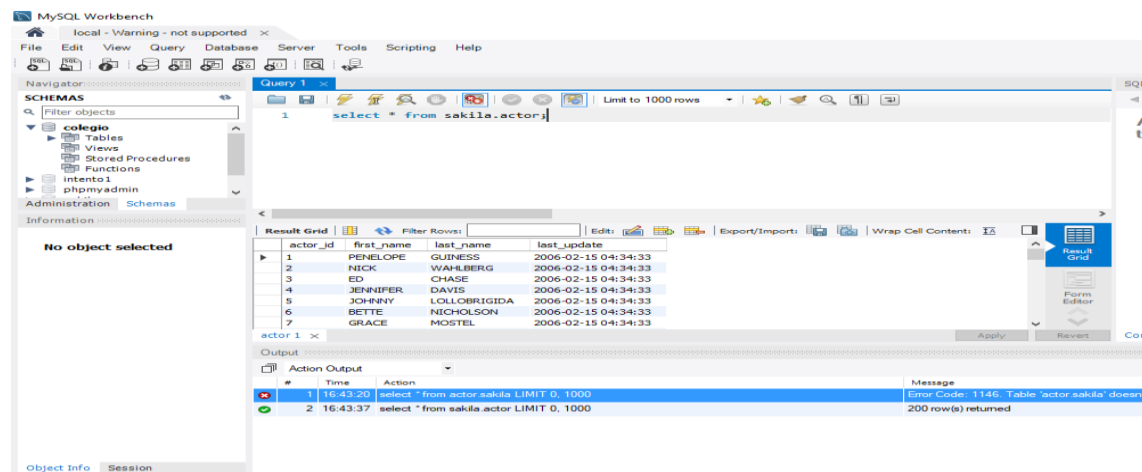
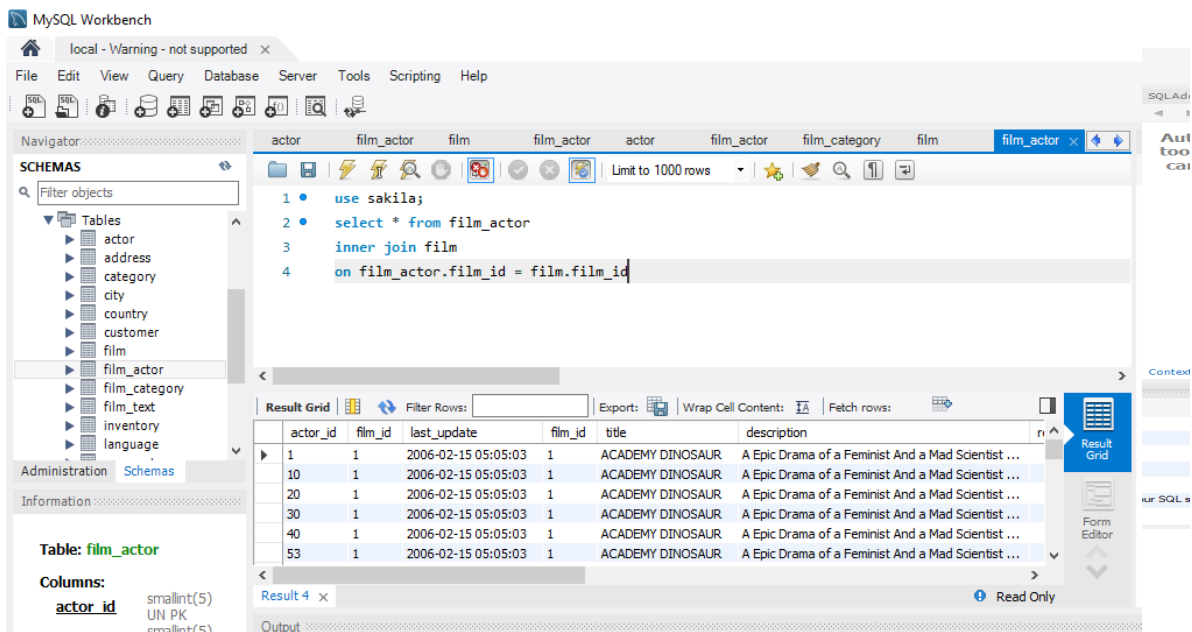


PRUEBA OBJETIVA PRACTICA FINAL

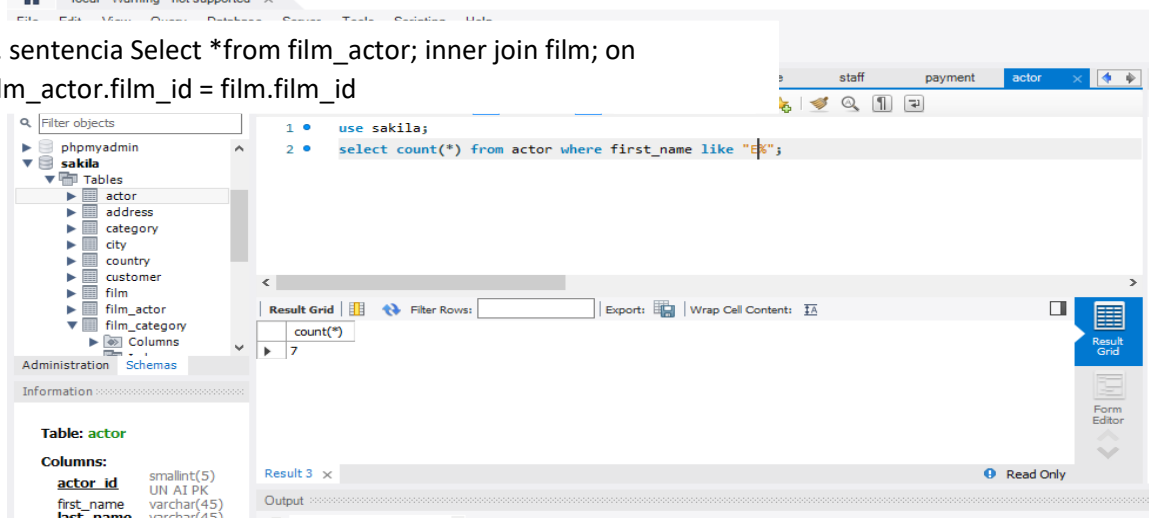
Realizar cinco consultas SQL de libre elección:



1. Sentencia SQL `Select * From`



4. sentencia `Select *from film_actor; inner join film; on film_actor.film_id = film.film_id`



3. Sentencia `Select count(*) from actor where first_name like "M%"`

MySQL Workbench

local - Warning - not supported

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

colegio

Tables

Views

Stored Procedures

Functions

intento1

phpmyadmin

sakila

Tables

actor

address

category

Administration Schemas

Information

Table: address

Columns:

address_id smallint(5)
UN AI PK

film film_actor actor film_actor film_category film film_actor store address

1 select * from address where district like "Texas"

Result Grid

address_id	address	address2	district	city_id	postal_code	phone	last_update
10	1795 Santiago de Compostela Way		Texas	295	18743	860452626434	2014-09-25 22:33:
122	333 Goinia Way		Texas	185	78625	909029256431	2014-09-25 22:33:
310	913 Coacalco de Berriozbal Loop		Texas	33	42141	262088367001	2014-09-25 22:33:
405	530 Lausanne Lane		Texas	135	11067	775235029633	2014-09-25 22:33:
567	1894 Boa Vista Way		Texas	178	77464	239357986667	2014-09-25 22:33:
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

address 2

Apply Revert

5. Sentencia select * from address where district like "Texas"

Realizar una subconsulta de libre elección.

MySQL Workbench

local - Warning - not supported

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

Foreign Keys

Triggers

film

film_actor

film_category

film_text

inventory

language

payment

rental

staff

Columns

Indexes

Administration Schemas

Information

Table: payment

Columns:

payment_id smallint(5)
UN AI PK

customer_id smallint(5)
FK

address rental rental inventory staff customer - Table customer address payment

1 select * from payment where rental_id > any

2 (select rental_id where amount >= 2.00)

Result Grid

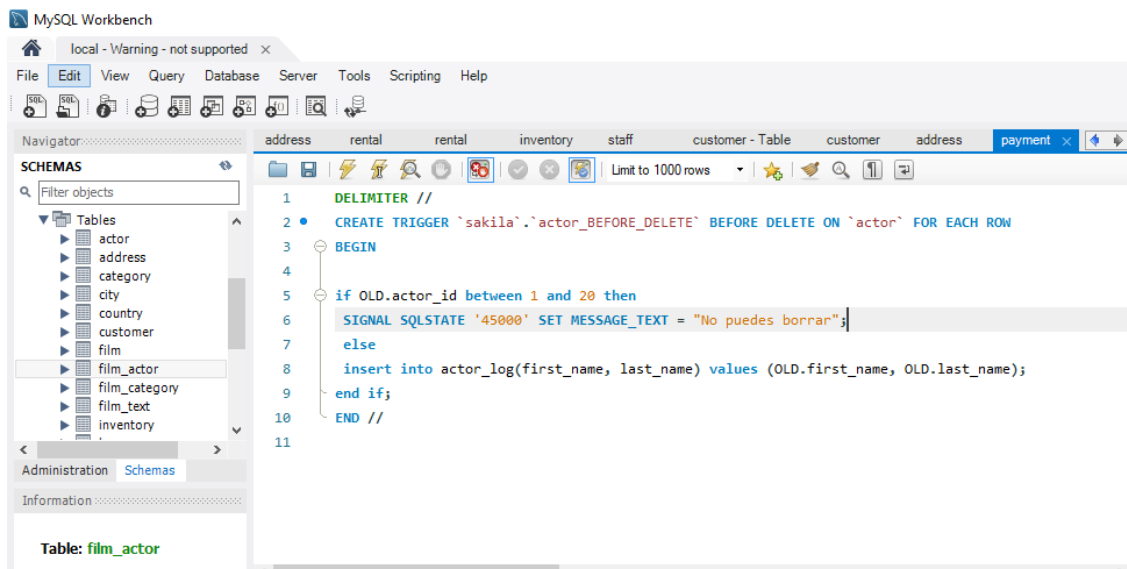
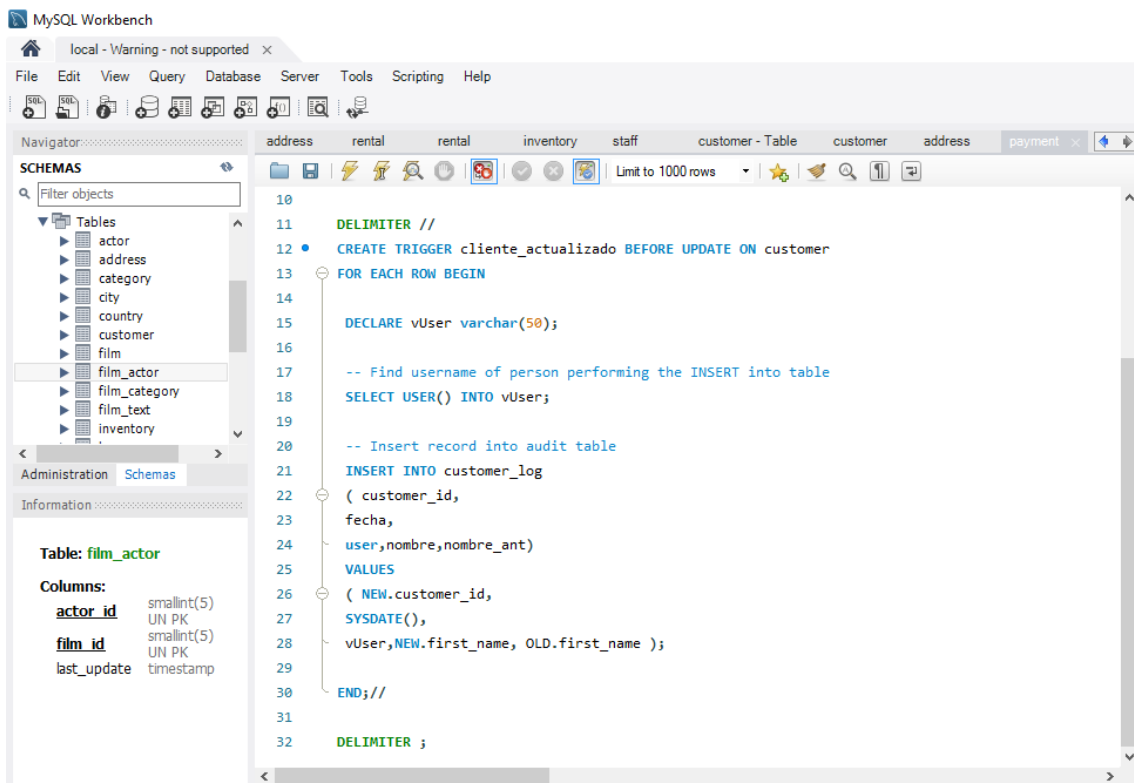
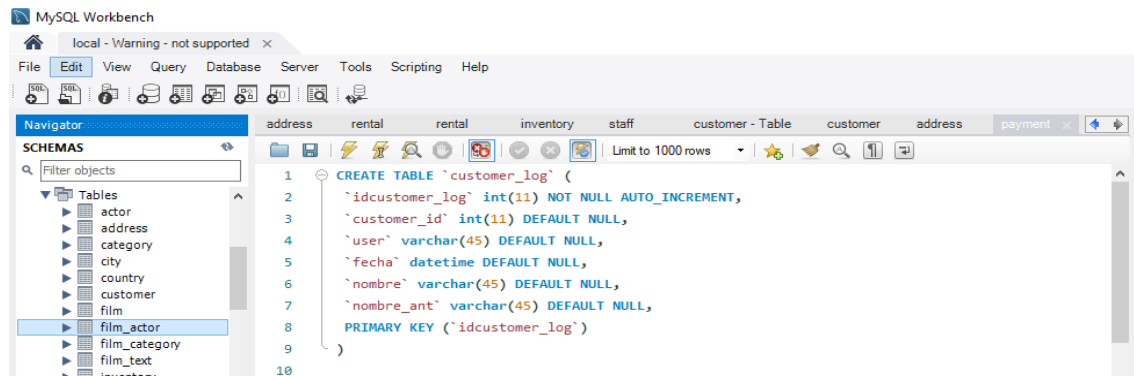
payment_id	customer_id	staff_id	rental_id	amount	payment_date	last_update
1	1	1	76	2.99	2005-05-25 11:30:37	2006-02-15 22:12:30
3	1	1	1185	5.99	2005-06-15 00:54:12	2006-02-15 22:12:30
5	1	2	1476	9.99	2005-06-15 21:08:46	2006-02-15 22:12:30
6	1	1	1725	4.99	2005-06-16 15:18:57	2006-02-15 22:12:30
7	1	1	2308	4.99	2005-06-18 08:41:48	2006-02-15 22:12:30
9	1	1	3284	3.99	2005-06-21 06:24:45	2006-02-15 22:12:30
10	1	2	4526	5.99	2005-07-08 03:17:05	2006-02-15 22:12:30

Result 2

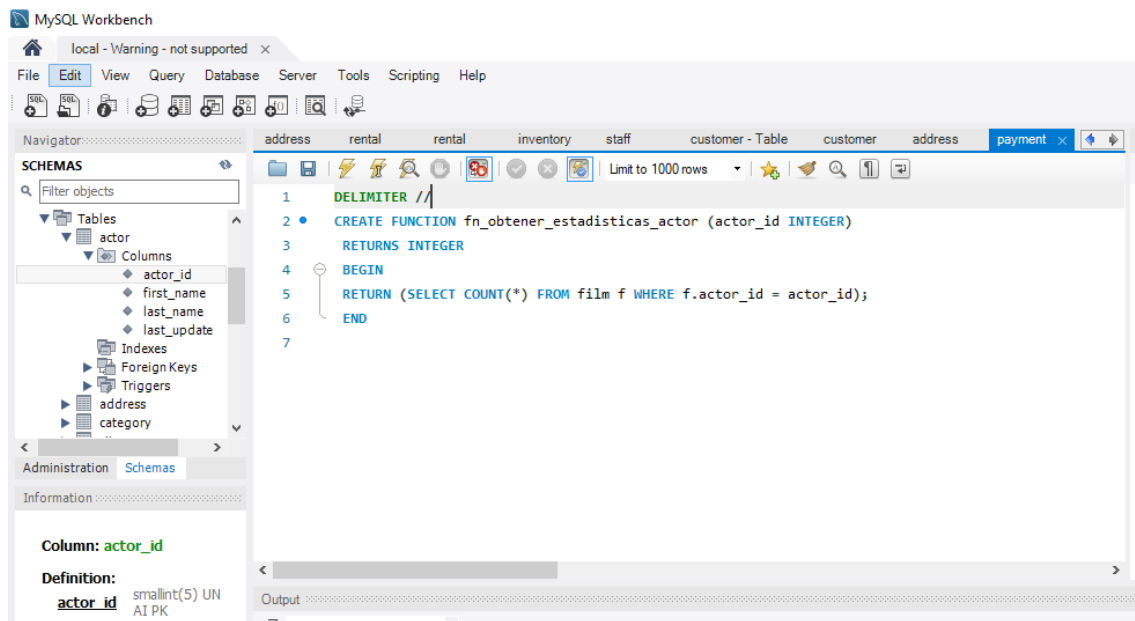
Read Only

Output

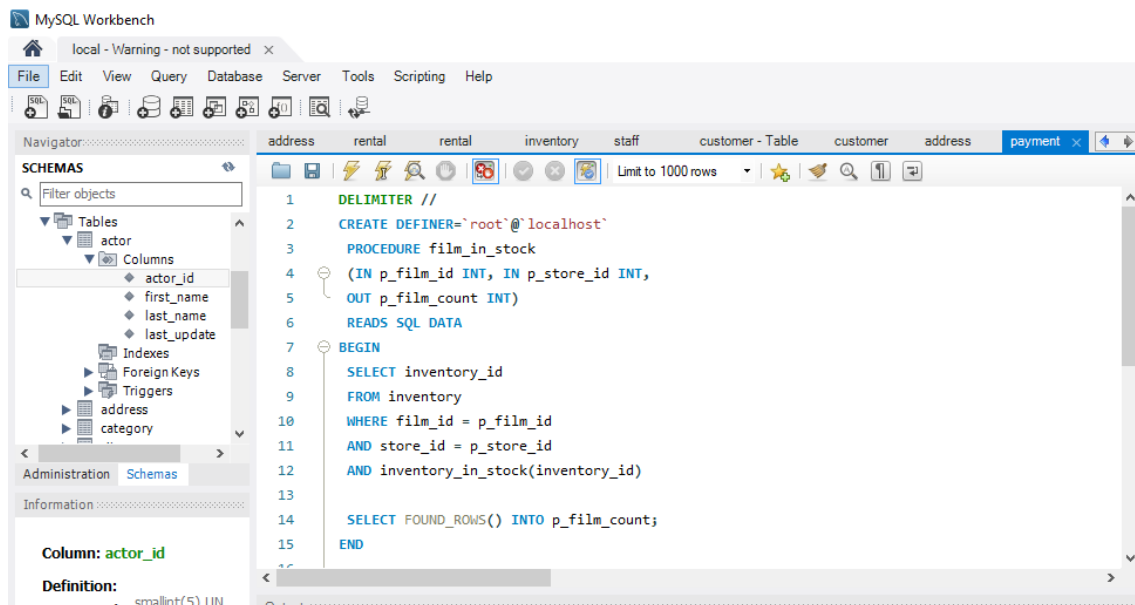
Implementar dos triggers.



Realiza una función.



Implementa un procedimiento.



MySQL Workbench

local - Warning - not supported x

File Edit View Query Database Server Tools Scripting Help

Navigator

Filter objects

SCHEMAS

- Tables
 - actor
 - Columns
 - actor_id
 - first_name
 - last_name
 - last_update
 - Indexes
 - Foreign Keys
 - Triggers
 - address
 - category

Administration Schemas

Information

Column: actor_id

Definition:

actor_id smallint(5) UNSIGNED AUTO_INCREMENT PRIMARY KEY

address rental rental inventory staff customer - Table customer address payment x

Limit to 1000 rows

```
14 SELECT FOUND_ROWS() INTO p_film_count;
15 END
16
17 call film_in_stock(123,1,@w)
18 select @w
19 DELIMITER //
20 CREATE DEFINER='root'@'localhost' PROCEDURE
21 `actores_por_categoria`
22 (in p_categoria varchar(50))
23 BEGIN
24 select distinct concat(first_name, ' ',last_name) actor
25 from actor join film_actor using (actor_id)
26 join film using (film_id)
27 join film_category using (film_id)
28 join category c using (category_id)
29 where c.name=p_categoria
30 order by actor;
31
32 END
```